AMENDMENTS TO THE CLAIMS

1. (Currently Amended) A method comprising:

storing native code associated with a first method within a native code space;

determining whether the ~~said~~ native code space exceeds a threshold in response to an invocation of a second method; and

reclaiming the ~~said~~ native code associated with the ~~said~~ first method and compiling byte code into native code associated with the ~~said~~ second method in response to the ~~said~~ determination.

2. (Currently Amended) The method as set forth in claim 2, wherein reclaiming the ~~said~~ native code associated with the ~~said~~ first method and compiling byte code into native code associated with the ~~said~~ second method in response to the ~~said~~ determination comprises reclaiming the ~~said~~ native code associated with the ~~said~~ first method in response to a determination that the ~~said~~ native code space exceeds the ~~said~~ threshold.

3. (Currently Amended) The method as set forth in claim 2, further comprising storing the ~~said~~ native code associated with the ~~said~~ second method within the ~~said~~ native code space in response to the ~~said~~ compilation.

4. (Currently Amended) The method as set forth in claim 2, further comprising:

invoking the ~~said~~ first method following the ~~said~~ reclamation; and

re-compiling byte code into the ~~said~~ native code associated with the ~~said~~ first method in response to the ~~said~~ invocation of the ~~said~~ first method.

5. (Currently Amended) The method as set forth in claim 2, wherein reclaiming the ~~said~~ native code associated with the ~~said~~ first method and compiling byte code into native code associated with the ~~said~~ second method in response to the ~~said~~ determination comprises compiling byte code into native code associated with the ~~said~~ second method ~~utilizing a JAVA virtual machine~~.

6. (Currently Amended) The method as set forth in claim 5, wherein compiling byte code into native code associated with the ~~said~~ second method ~~utilizing a JAVA virtual machine~~ comprises compiling byte code into native code associated with the ~~said~~ second method utilizing a just-in-time compiler.

7. (Currently Amended) The method as set forth in claim 2, wherein reclaiming the ~~said~~ native code associated with the ~~said~~ first method and compiling byte code into native code associated with the ~~said~~ second method in response to the ~~said~~ determination comprises~~:~~ determining whether the ~~said~~ first method is active or inactive; and reclaiming the ~~said~~ native code associated with the ~~said~~ first method in response to a determination that the ~~said~~ first method is inactive.

8. (Currently Amended) The method as set forth in claim 7, wherein:

reclaiming the ~~said~~ native code associated with the ~~said~~ first method and compiling byte code into native code associated with the ~~said~~ second method in response to the ~~said~~ determination further comprises determining whether the ~~said~~ first method is hot or cold in response to a determination that the ~~said~~ first method is inactive[[,]]; and

reclaiming the ~~said~~ native code associated with the ~~said~~ first method in response to a determination that the ~~said~~ first method is inactive comprises reclaiming the ~~said~~ native code associated with the ~~said~~ first method in response to a determination that the ~~said~~ first method is cold.

9. (Currently Amended) A data processing system-readable medium having a plurality of instructions executable by a data processing system embodied therein, wherein the ~~said~~ plurality of instructions when executed cause the ~~said~~ data processing system to perform operations comprising:

storing native code associated with a first method within a native code space;

determining whether the ~~said~~ native code space exceeds a threshold in response to an invocation of a second method; and

reclaiming the ~~said~~ native code associated with the ~~said~~ first method and compiling byte code into native code associated with the ~~said~~ second method in response to the ~~said~~ determination.

10. (Currently Amended) The data processing system-readable medium of claim 9, wherein reclaiming the ~~said~~ native code associated with the ~~said~~ first method and compiling byte code into native code associated with the ~~said~~ second method in response to the ~~said~~ determination comprises reclaiming the ~~said~~ native code associated with the ~~said~~ first method in response to a determination that the ~~said~~ native code space exceeds the ~~said~~ threshold.

11. (Currently Amended) The data processing system-readable medium of claim 9, wherein the ~~said~~ plurality of instructions when executed further cause the ~~said~~ data processing system to perform operations comprising storing the ~~said~~ native code associated with the ~~said~~ second method within the ~~said~~ native code space in response to the ~~said~~ compilation.

12. (Currently Amended) The data processing system-readable medium of claim 9, wherein the ~~said~~ plurality of instructions when executed further cause the ~~said~~ data processing system to perform operations comprising invoking the ~~said~~ first method following the ~~said~~ reclamation; and

re-compiling byte code into the ~~said~~ native code associated with the ~~said~~ first method in response to the ~~said~~ invocation of the ~~said~~ first method.

13. (Currently Amended) The data processing system-readable medium of claim 9, wherein reclaiming the ~~said~~ native code associated with the ~~said~~ first method and compiling byte code into native code associated with the ~~said~~ second method in response to the ~~said~~ determination comprises compiling byte code into native code associated with the ~~said~~ second method ~~utilizing a JAVA virtual machine~~.

14. (Currently Amended) The data processing system-readable medium of claim 13, wherein compiling byte code into native code associated with the ~~said~~ second method ~~utilizing a JAVA virtual machine~~ comprises compiling byte code into native code associated with the ~~said~~ second method utilizing a just-in-time compiler.

15. (Currently Amended) The data processing system-readable medium of claim 9, wherein reclaiming the ~~said~~ native code associated with the ~~said~~ first method and compiling byte code into native code associated with the ~~said~~ second method in response to the ~~said~~ determination comprises~~:~~ determining whether the ~~said~~ first method is active or inactive; and reclaiming the ~~said~~ native code associated with the ~~said~~ first method in response to a determination that the ~~said~~ first method is inactive.

16. (Currently Amended) The data processing system-readable medium of claim 15, wherein:

reclaiming the ~~said~~ native code associated with the ~~said~~ first method and compiling byte code into native code associated with the ~~said~~ second method in response to the ~~said~~ determination further comprises determining whether the ~~said~~ first method is hot or cold[[,]]; and

reclaiming the ~~said~~ native code associated with the ~~said~~ first method in response to a determination that the ~~said~~ first method is inactive comprises reclaiming the ~~said~~ native code associated with the ~~said~~ first method in response to a determination that the ~~said~~ first method is cold.

17. (Currently Amended) A data processing system comprising:

a storage device;

a processor coupled with the storage device, the processor to process data and execute instructions; and

a memory coupled with the storage device and the processor, the memory to store data including a plurality of instructions which when executed by the ~~said~~ processor cause the ~~said~~ data processing system to perform operations ~~comprising:~~having

storing native code associated with a first method within a native code space of the ~~said~~ memory[[;]];

determining whether the ~~said~~ native code space exceeds a threshold in response to an invocation of a second method[[;]]; and

reclaiming the ~~said~~ native code associated with the ~~said~~ first method and compiling byte code into native code associated with the ~~said~~ second method in response to the ~~said~~ determination.

18. (Currently Amended) The data processing system of claim 17, wherein reclaiming the ~~said~~ native code associated with the ~~said~~ first method and compiling byte code into native code associated with the ~~said~~ second method in response to the ~~said~~ determination comprises reclaiming the ~~said~~ native code associated with the ~~said~~ first method in response to a determination that the ~~said~~ native code space exceeds the ~~said~~ threshold.

19. (Currently Amended) The data processing system of claim 17, wherein the ~~said~~ plurality of instructions when executed further cause the ~~said~~ data processing system to perform operations comprising storing the ~~said~~ native code associated with the ~~said~~ second method within the ~~said~~ native code space in response to the ~~said~~ compilation.

20. (Currently Amended) The data processing system of claim 17, wherein the ~~said~~ plurality of instructions when executed further cause the ~~said~~ data processing system to perform operations comprising invoking the ~~said~~ first method following the ~~said~~ reclamation; and

re-compiling byte code into the ~~said~~ native code associated with the ~~said~~ first method in response to the ~~said~~ invocation of the ~~said~~ first method.

21. (Currently Amended) The data processing system of claim 17, wherein reclaiming the ~~said~~ native code associated with the ~~said~~ first method and compiling byte code into native code associated with the ~~said~~ second method in response to the ~~said~~ determination comprises compiling byte code into native code associated with the ~~said~~ second method ~~utilizing a JAVA virtual machine~~

22. (Currently Amended) The data processing system of claim 21, wherein compiling byte code into native code associated with the ~~said~~ second method ~~utilizing a JAVA virtual machine~~ comprises compiling byte code into native code associated with the ~~said~~ second method utilizing a just-in-time compiler.

23. (Currently Amended) The data processing system of claim 17, wherein reclaiming the ~~said~~ native code associated with the ~~said~~ first method and compiling byte code into native code associated with the ~~said~~ second method in response to the ~~said~~ determination comprises: determining whether the ~~said~~ first method is active or

inactive; and reclaiming the ~~said~~ native code associated with the ~~said~~ first method in response to a determination that the ~~said~~ first method is inactive.

24. (Currently Amended) The data processing system of claim 23, wherein:

reclaiming the ~~said~~ native code associated with the ~~said~~ first method and compiling byte code into native code associated with the ~~said~~ second method in response to the ~~said~~ determination further comprises determining whether the ~~said~~ first method is hot or cold, and

reclaiming the ~~said~~ native code associated with the ~~said~~ first method in response to a determination that the ~~said~~ first method is inactive comprises reclaiming the ~~said~~ native code associated with the ~~said~~ first method in response to a determination that the ~~said~~ first method is cold.

25. (New) An apparatus comprising:

a processor to process data and execute instructions; and

a memory coupled with the processor, the memory to store data including a plurality of instructions which when executed by the processor cause the data processing system to perform operations including

storing native code associated with a first method within a native code space of the memory,

determining whether the native code space exceeds a threshold in response to an invocation of a second method, and

reclaiming the native code associated with the first method and compiling byte code into native code associated with the second method in response to the determination.

26. (New) The apparatus of claim 25, wherein reclaiming the native code associated with the first method and compiling byte code into native code associated with the second method in response to the determination includes reclaiming the native code associated with the first method in response to a determination that the native code space exceeds the threshold.

27. (New) The apparatus of claim 25, wherein the plurality of instructions when executed further cause the data processing system to perform operations including storing the native code associated with the second method within the native code space in response to the compilation.

28. (New) The apparatus of claim 25, wherein the plurality of instructions when executed further causes the data processing system to perform operations including

invoking the first method following the reclamation, and

re-compiling byte code into the native code associated with the first method in response to the invocation of the first method.